REMARKS

Claims 1-18 are pending in the present application. Claims 8, 9, 14, and 15 were amended to correct minor typographical errors. No new matter was added. Reconsideration of the claims is respectfully requested.

I. Interview Summary

Examiner Coffy and Theodore D. Fay III discussed the objection to the specification. No agreement was reached.

II. Specification

The examiner states that, "the abstract... should be within the range of 50 to 150 words." Applicants have amended the abstract so that it is now 140 words. No new matter was added.

III. 35 U.S.C. § 103, Obviousness

The examiner has rejected claims one through eighteen under 35 U.S.C. § 103 as being unpatentable over Nareddy et al. Method and System for Parsing Navigation Information, U.S. Patent 6,785,666 (Aug. 31, 2004) in view of Nair et al. System and Method for Efficient and Adaptive Web Access Filtering, U.S. Patent 6,741,990 (May 25, 2004). This rejection is respectfully traversed.

Regarding claims one through eighteen, the examiner states:

> Nareddy teaches the invention substantially as claimed including a method and system for providing customers with access data to and analysis of event data, which may be stored in log files. A warehouse server receives customer data; it converts the customer data into a format that is more conductive to processing by decision support system applications by which customers can analyze their data. (See abstract).

(Office Action of September 21, 2004, page 2)

The examiner misapprehends Nareddy and as a result fails to state a prima facie case of obviousness. Nareddy shows a method of providing customers with access to

data from web logs. Nareddy also shows a method of reformatting the data in order allow a *support system application* to more easily use the data. Nareddy states:

> The parser analyzes the low-level events of the customer data and identifies high-level events and converts the customer data into a format that facilitates processing by the decision support system applications.

Nareddy, col. 6, ll. 52-55.

Thus, Nareddy reformats web log data for other applications to use. However, Nareddy is of no help if the web log data cannot be read in the first place. If the web server in Nareddy attempted to contact a web server having a different access log formats, then Nareddy could not access the web log data in the first place. Nareddy is explicit about this fact:

> The parser includes a filter log entry component 311, a normalize log entry component 312, a generate dimensions component 313, an identify sessions component 314, and a generate aggregate statistics component 315. The filter log entry component identifies which log entries should not be included in the main data warehouse. *For example, a log entry that has an invalid format should not be included.* The normalize log entry component normalizes the data in a log entry.

Nareddy, col. 7, ll. 13-23 (emphasis added)

Nareddy explicitly provides that a log entry having an invalid format should not be included in the filtered log. If the Nareddy server accessed another server having a different log format, then the format would be invalid and Nareddy would exclude entirely all information from the other server. The last sentence in the above quote shows that normalization of the data in the log entry only occurs once the log has been filtered. Thus, all reference Nareddy makes to reformatting information in the log has nothing to do with translating a web log format into a common web log format as claimed.

Claim one provides as follows:

> 1. (original) A method for establishing compatibility between heterogeneous web server access log formats, comprising:
> supplying a description of an access log file of a web server, by:
> opening a customizable configuration file;
> if the access log is static, setting a log pattern definition to describe data elements, order, and syntax of log entries;

if the access log is dynamic, setting a dictionary feature for a log pattern definition;
saving and exiting the configuration file; and
invoking a computer process, wherein the process in turn invokes a web server access log translation engine (WSALTE) which translates the described web server access log file to a desired log format and returns the translated file back to the computer process.

Claim one clearly claims a method for establishing compatibility between heterogeneous web server access log formats. As explained above, Nareddy is incapable of doing so and is devoid of disclosure regarding any of the claim limitations. Nair, which shows a method of excluding IP addresses from a web log that can already be read, fails to cure the lack of disclosure in Nareddy. Thus, the proposed combination cannot result in the claimed inventions. Accordingly, the examiner has failed to state prima facie obviousness rejections.

Now turning to the individual rejections, the examiner rejects claim 1 under the assertion that:

Claim 1:
Referring to claim 1, Nareddy teaches a method for establishing compatibility between heterogeneous web server access log formats, comprising: (See col. 1 and 2).
supplying a description of an access log file of a web server, by: (See col. 4, lines 1-16)
opening a customizable configuration file;
if the access log is static, setting a log pattern definition to describe data elements, order, and syntax of log entries; (See col. 5, lines 5-8).
if the access log is dynamic, setting a dictionary feature for a log pattern definition; (See col. 6, lines 42-56).
saving and exiting the configuration file; and (See col. 5, lines 21-24).
invoking a computer process, wherein the process in turn invokes a web server access log translation engine (WSALTE) which translates the described web server access log file to a desired log format and returns the translated file back to the computer process. (See col. 6, lines 42-56).
Nareddy teaches a method and system for parsing navigation information stored in access log files. Nareddy does not specifically disclose the concept of a dynamic log file. However, Nair specifically teaches a dynamic log file where blocks of IP addresses are to be filtered. (See col. 6, lines 27-65).

> Hence, it would have been obvious at the time of the invention for an artisan of ordinary skill in the art to combine the translation of different access log files taught by Nareddy with the dynamic algorithm as disclosed by Nair. Such a system would allow a user to understand the effectiveness of a web site. Therefore, claim 1 is rejected.

(Office Action of September 21, 2004, page 3)

As shown above, Nareddy does not show any of the claim limitations and Nair fails to cure the lack of disclosure; accordingly, the examiner has failed to state prima facie obviousness rejections. The examiner cites various sections of Nareddy, though as shown below the examiner misapprehends Nareddy.

For the proposition that Nareddy shows a method for establishing compatibility between heterogeneous web server access log formats, the examiner cites columns one and two of Nareddy. However, the cited text does not discuss this problem. Given the extensive text within those two columns, Applicants request that the examiner point out the exact relevant text or allow the claims.

For the proposition that Nareddy shows opening a customizable configuration file and if the access log is static, setting a log pattern definition to describe data elements, order, and syntax of log entries, the examiner cites from Nareddy as follows:

> These categories may be stored in a categoly [sic] dimension table. Also, certain facts, such as the collection of log entries that comprise a single user web access session or visit, may only to be derivable by analyzing a series of log entries.

Nareddy, col. 5, ll. 5-8.

The cited text is plainly does not describe any of the claimed limitations. Log entries may be analyzed by Nareddy, though if the log formats were incompatible in the first place, then Nareddy's proposals would be meaningless.

For the proposition that Nareddy shows the limitation of "if the access log is dynamic, setting a dictionary feature for a log pattern definition," the examiner cites from Nareddy as follows:

> The pitcher is responsible for retrieving instructions from the data warehouse server, collecting the customer data in accordance with the retrieved instructions, and uploading the customer data to the data warehouse server. The monitor is responsible for monitoring the operation of the pitcher and detecting when the pitcher may

> have problems in collecting and uploading the customer data. When the monitor detects that a problem may occur, it notifies the data warehouse server so that corrective action may be taken in advance of the collecting and uploading of the customer data. For example, the pitcher may use certain log on information (e.g., user ID and password) to access a customer web server that contains customer data to be uploaded.

Nareddy, col. 6, ll. 17-29.

Again, the cited text plainly does not describe the claimed limitation. Nowhere does Nareddy describe setting a dictionary feature for a log pattern definition.

For the proposition that Nareddy shows saving and exiting the configuration file, the examiner cites Nareddy as follows:

> The parser may use parser configuration data that defines, on a customer-by-customer basis, the high-level data to be derived from the log entries. For example, the parser configuration data may specify the mapping of URIs to web page categories.

Nareddy, col. 5, ll. 21-25.

Again, the cited text plainly does not describe the claimed limitation. Nowhere does Nareddy describe setting a dictionary feature for a log pattern definition.

For the proposition that Nareddy shows the invoking limitation as claimed, the examiner cites Nareddy as follows:

> The data receiver component of the data warehouse server includes a status receiver sub-component 271, a catcher sub-component 272, an FTP server 273, a status database 274, and a collected data database 275. The status receiver receives status reports from the customer servers and stores the status information in the status database. The catcher receives and processes the customer data that is uploaded from the customer web sites and stores the data in the collected data database. The data processor component includes a parser sub-component 281 and a loader sub-component 282. *The parser analyzes the low-level events of the customer data and identifies high-level events and converts the customer data into a format that facilitates processing by the decision support system applications.*

Nareddy, col. 6, ll. 42-56 (emphasis added).

Again, Nareddy plainly does not show the claimed limitation regarding a web server access log translation engine. Although Nareddy does discuss converting customer data into a format that facilitates processing by the decision support system applications, Nareddy does not describe translating the web log files in the first place.

Nareddy merely reformats information contained in web logs that have already been read in the first place. Thus Nareddy does not show the claimed limitations.

As shown above, none of the text cited by Nareddy shows or suggests the claimed limitations. Nair fails to cure the lack of disclosure in Nareddy. Accordingly, the examiner has failed to state prima facie obviousness rejections.

In addition, the examiner cites Nair for the proposition that Nair teaches a dynamic log file where blocks of IP addresses are to be filtered, citing Nair as follows:

> Given the list of exclusion IP addresses to filter, an optimum algorithm needs to be selected for searching in the list of exclusion IP addresses for matches. As mentioned previously, the list of accesses to be filtered, or the list of exclusion IP address in this case, often varies from one client web server to another because of different performance analysis or different filtering requirements exist at different client web servers. For example, each client may decide to filter activities coming from its own company, so the list of exclusion IP address in each case is different. To reduce the cost of identifying whether a particular IP address in the web log file is to be ignored, the adaptive process explores the performance of several different algorithms and data structures. In block P310, metrics are generated for each algorithm to calculate the effectiveness of the algorithm for the list of exclusion IP address. The algorithms is chosen from any number of predefined methods, including, but not limited to, binary search, multi-level dynamic table indexing, adaptive hashing, and bit pattern based exclusion.
>
> The interaction of these algorithms with the IP addresses from the web log file (block P200) will be explained in more detail later, as the discussion now centers on computing the metrics for each algorithm. In one implementation, the metrics to be used to identify the optimum algorithm for a list of exclusion IP addresses are the number of exclusion IP addresses to be filtered and the number of unique combinations to be filtered. For example, binary search algorithm is often employed if there is a relative small number of individual exclusion IP addresses to filter. However, binary search algorithm is probably not the best algorithm if you have blocks of IP addresses to filter. In case of binary search, the number of exclusion IP addresses to filter may be one of the determinative metrics, and as the number grows the performance decreases.
>
> Multi-level dynamic table indexing algorithm is suitable when there are blocks of IP addresses to filter. The metric in this case would be the number of unique combinations of octets in the list of

> exclusion IP addresses, assuming that eight bits are used for indexing.

The cited text plainly does not teach a using a dictionary definition if the access log is dynamic. Although Nair does discuss multi-level dynamic table indexing, he does so in the context of creating algorithms for excluding IP addresses from a web log. Like Nareddy, Nair's methods depend on first successfully reading the web log in the first place. Nair's methods are meaningless unless this is step is performed first. The claimed method is directed to translating incompatible web logs and is thus completely distinct from Nairs's methods. Given that both Nareddy and Nair fail to show any of the claimed limitations, the examiner has failed to state prima facie obviousness rejections.

In addition, the examiner has failed to provide a motivation to combine the references. The examiner states that:

> Hence, it would have been obvious at the time of the invention for an artisan of ordinary skill in the art to combine the translation of different access log files taught by Nareddy with the dynamic algorithm as disclosed by Nair. Such a system would allow a user to understand the effectiveness of a web site.

(Office Action of September 21, 2004, page 3)

This statement provides no reason to motivate one of ordinary skill to combine the references. Thus, the examiner again has failed to state a prima facie obviousness rejection. In the light of the fact that neither Nair nor Nareddy show any of the claimed limitations, no such motivation can exist. Accordingly, the claims are non-obvious in view of the cited references.

In addition, the examiner's statement makes no sense in the context of the claimed invention. The examiner states that, "Such a system would allow a user to understand the effectiveness of a website." This statement is irrelevant to the claimed method of establishing compatibility between heterogeneous web sites because understanding the effectiveness of the website has nothing to do with establishing compatibility. The fact that the statement makes no sense in the context of the claimed invention shows that the examiner has failed to establish a proper motivation to combine the references. Accordingly, the examiner has failed to state prima facie obviousness rejections.

In addition, one of ordinary skill would have no reason to combine Nareddy and Nair in the manner proposed in the first place. Nareddy is directed to accessing and formatting data that has already been accessed from a web log. Nair is directed to filtering out IP addresses from a web log. When read as a whole, the two reference have nothing to do with the claimed inventions. Thus, one of ordinary skill would be motivated to ignore or avoid the references when dealing with the problems addressed by the claimed inventions. Accordingly, the claims are non-obvious.

Regarding claims two through five, these dependent claims are non-obvious in view of the cited references for the same reasons that claim one is non-obvious in view of the cited references. In addition, claims two through five contain additional patentable features not taught or suggested by the combination of Nareddy and Nair. For example, neither reference shows "if a data element contains a delimiter that may exist in another data element, isolating the data by replacing the WSALTE terminology with a user substitute definition," as claimed in claim three. Although the examiner cites text from Nareddy in support of the assertion that Nareddy teaches the claimed limitation, as shown above the examiner has misapprehended Nareddy. Similarly, the limitations of the other dependent claims are not taught or suggested by Nareddy and the lack of disclosure in Nareddy is not cured by Nair. Hence, the examiner has failed to state prima facie obviousness rejections of claims two through five. Moreover, for similar reason to those given with respect to claim one, no one would be motivated to combine or modify Nareddy and Nair to achieve the claimed inventions. Accordingly, claims two through five are non-obvious.

In addition, the examiner's reasoning with respect to dependent claims two through five makes no sense. For example, the examiner states that, with respect to claim two, "it would have been obvious.... to use the translation of different access log files taught by Nareddy." As discussed above, Nareddy does not show the translation of different access log files. Hence, the examiner's reasoning that the claim would have been obvious makes no sense.

Regarding claims six through eighteen, the examiner rejects the claims under the same reasoning given with respect to claims one through five. For the same reasons given above, claims six through eighteen are also non-obvious over the cited references

and the examiner has failed to state prima facie obviousness rejections of claims six through eighteen. Therefore, the rejection of claims one through 18 under 35 U.S.C. § 103 has been overcome.

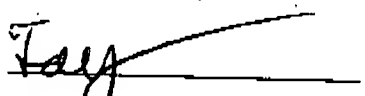
IV. Conclusion

It is respectfully urged that the subject application is patentable over Nareddy et al. (US 6,785,666) in view of Nair et al. (US 6,741,990.) and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: December 21, 2004

Respectfully submitted,



Theodore D. Fay III
Reg. No. 48,504
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicant